RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

Géosciences pour une Terre durable
brgm

COMMENT MONOPOLISER LES RESSOURCES DU CLUSTER REGIONAL

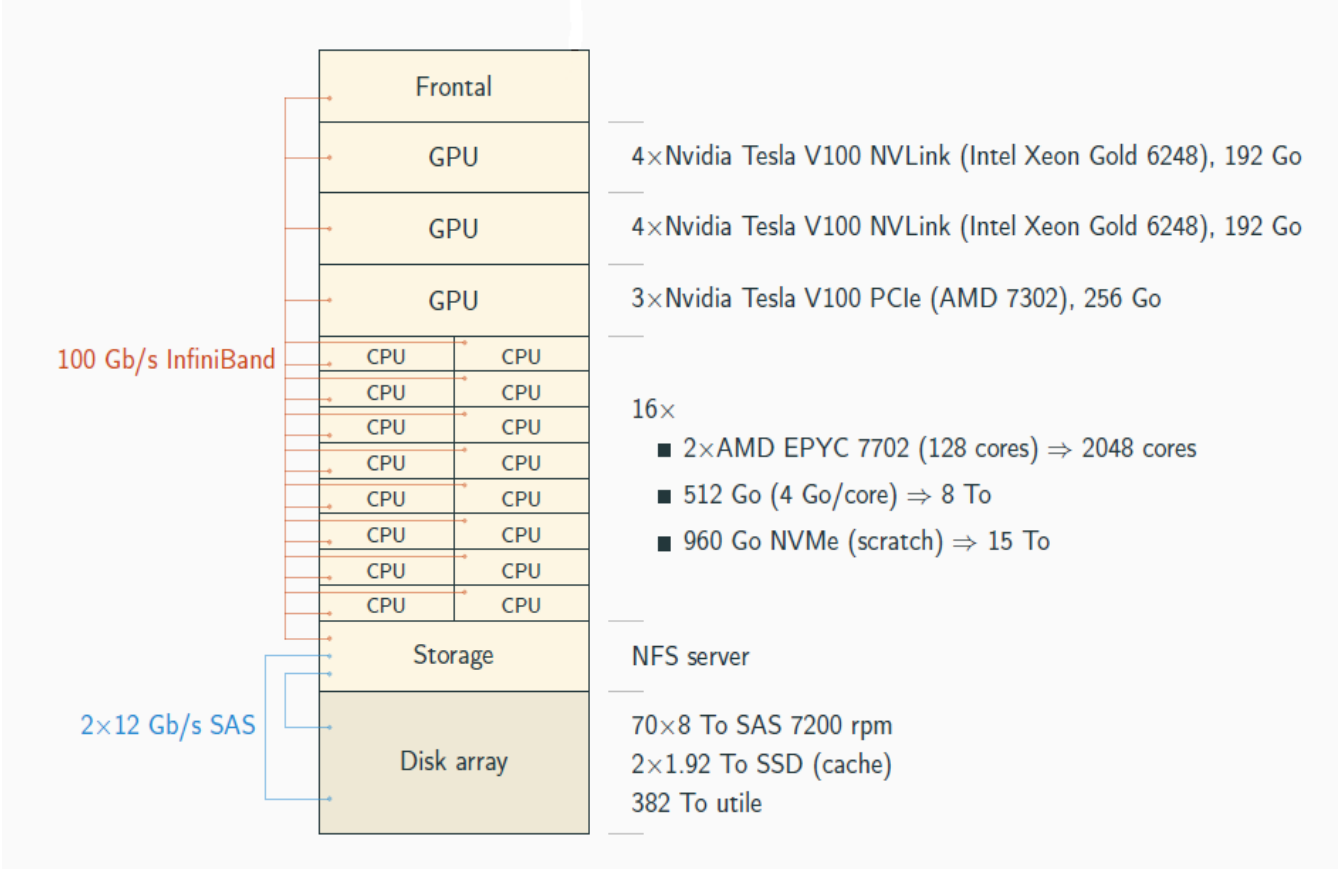
Cluster Régional

Ressources de calcul mutualisées à l'échelle de la région Centre Val de Loire

leto.cascimodot.datacentre-valdeloire.fr

- Ressources de calcul HPC IA visualisation :
  - CPU
  - GPU
  - Stockage

- Environnements logiciels pré-installés mutualisés



Cluster Régional

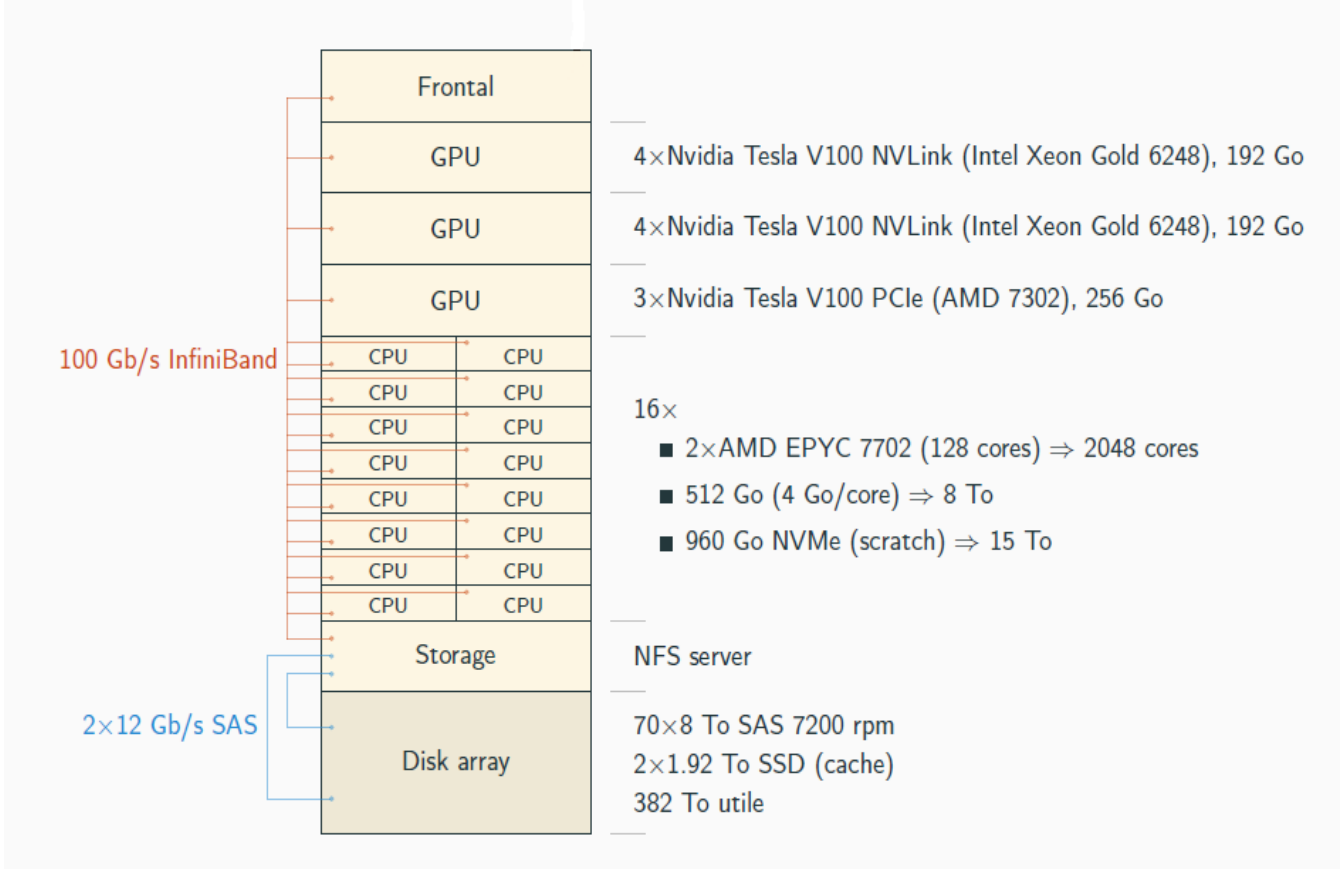
Ressources de calcul mutualisées à l'échelle de la région Centre Val de Loire

leto.cascimodot.datacentre-valdeloire.fr

- Au 1er décembre 2021 :
  - 23 laboratoires/établissements
  - 146 comptes
  - Plus de 23 000 jobs depuis la mise en service

→ nécessité d'utiliser un ordonnanceur pour
  - la répartition des ressources entre les différents jobs et utilisateurs
  - le cloisonnement des ressources (CPU/GPU)
  - la supervision des ressources

SLURM



SLURM : principe

Pour bénéficier de ressources vous devez décrire votre job

Un job

- Ensemble de tâches à réaliser nécessitant :
  - Du CPU
    - 1 cœur : job séquentiel
    - plusieurs cœurs : job parallèle
      - Toutes les tâches partagent une zone mémoire sur une même machine : job multithreadé ou à mémoire partagée (cf. OpenMP)
      - Les tâches s'échangent des informations entre elles via une librairie : job parallèle à mémoire distribuée (cf. MPI)
  - De la mémoire
  - Des GPU
  - …En une durée donnée

Une ressource de calcul

- Un nœud de calcul : une machine physique est composée de plusieurs processeurs contenant eux-mêmes des cœurs physiques

Schématiquement

- *Le cluster est un plateau de Tetris*
- *Les jobs sont des pièces à 3 dimensions (nombre de cœurs, quantité de mémoire, durée)*
- *SLURM est le joueur qui cherche à optimiser le placement sur le plateau dynamiquement*

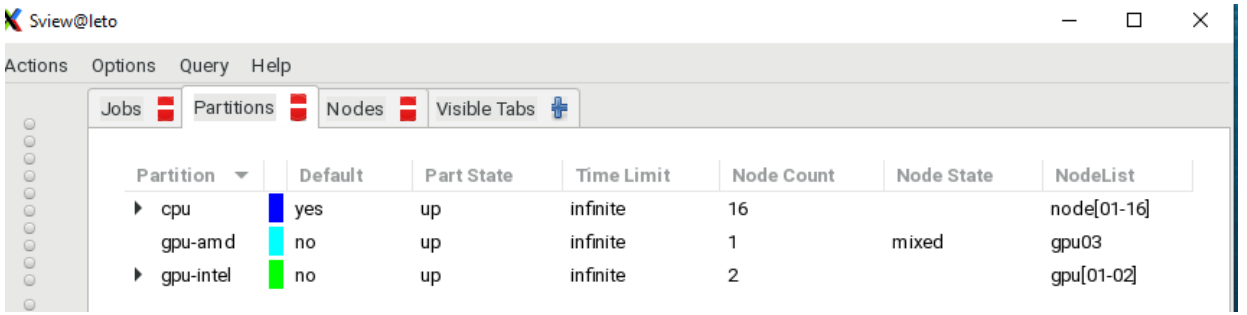
Pour utiliser au mieux les ressources vous devez connaitre votre application

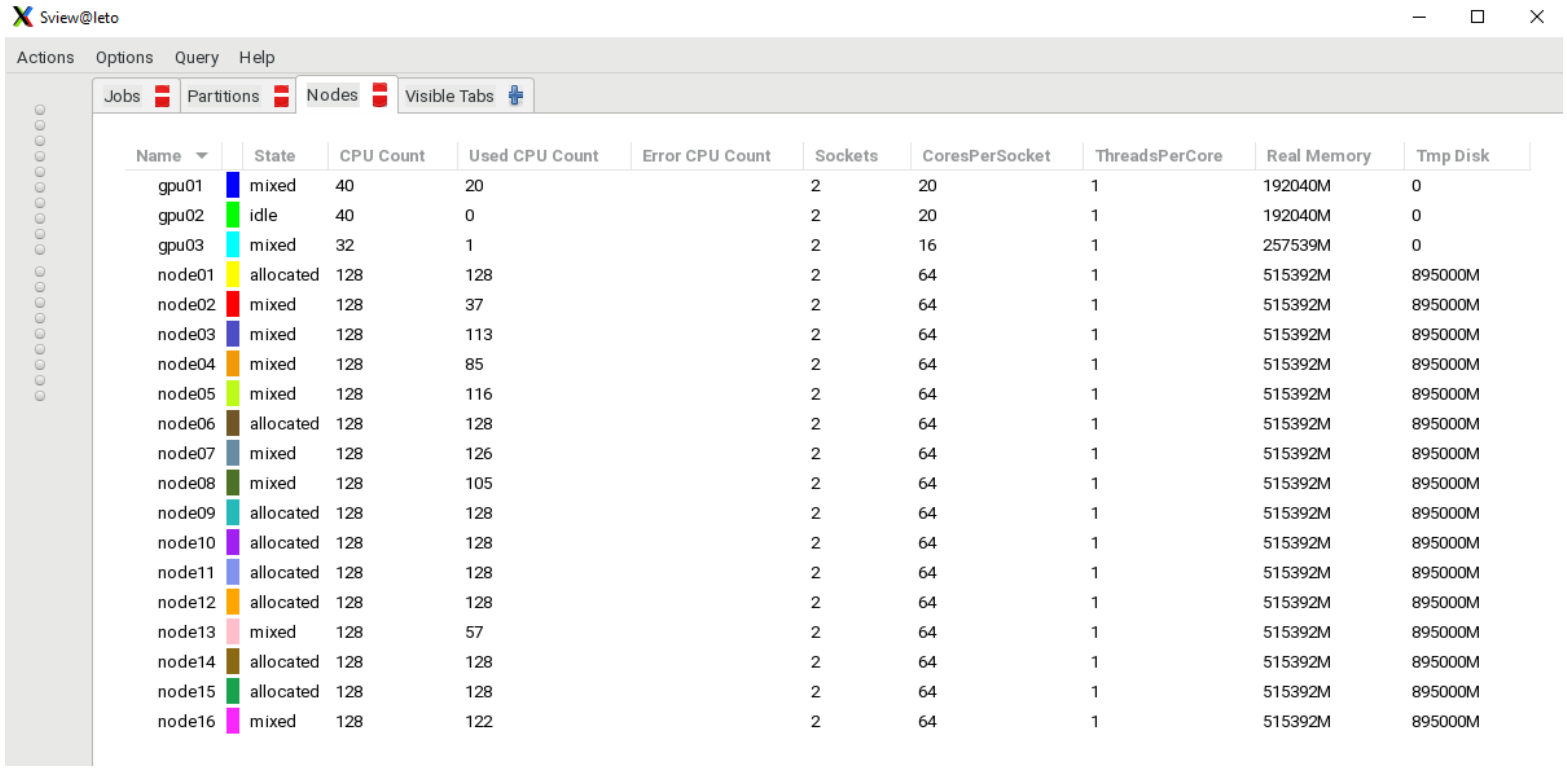
Etat du cluster

- sview
  - Jobs
  - Partitions : regroupement des nœuds ayant les mêmes caractéristiques physiques ou contraintes
  - Nœuds

- gpu_in_use

```
fboulahya@leto:~/JourneeCascimodot> gpu_in_use
queue     gpu_in_use      gpu_avail
---------------------------------
gpu-intel         1       7
gpu-amd           0       3
```

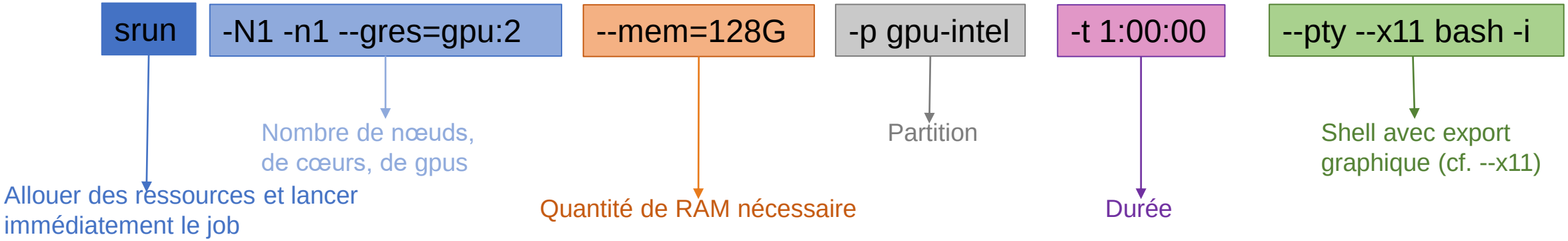




TRAVAIL EN INTERACTIF

Si vous ne précisez aucune ressource, un cœur avec 2Gb de RAM pendant 12 heures sur la partition cpu vous sera alloué.

- Pour quel usage
  - Compilation
  - Installation de packages R/python
  - Test
  - Exploration/Visualisation de données
  - Jobs qui nécessitent de l'interactivité (saisie utilisateur)
- Quelle commande :



TRAVAIL EN BATCH

- Pour quel usage
  - Compilation
  - Simulation
  - Notebooks
  - …

- Comment :
  - Description du job dans un fichier
  - Toutes les lignes commençant par #SBATCH seront interprétées par SLURM pour spécifier les ressources du job (cf. mêmes options que srun)
  - On soumet ce fichier à SLURM avec la commande sbatch

TRAVAIL EN BATCH

Exemples

- Indiquer une durée à vos jobs!

```
#!/bin/bash
### Script de soumission SLURM

#SBATCH -J JOB_SEQ
#SBATCH -p cpu
#SBATCH -N 1 -n 1
#SBATCH -t 0:30:00


module load gcc/10.2

./sequentiel.sh
```

```
#!/bin/bash
### Script de soumission SLURM

#SBATCH -J JOB_MULTITHREAD
#SBATCH -p cpu
#SBATCH --ntasks=1 --cpus-per-task=16
#SBATCH -t 0:30:00

#A decommenter pour jobs OpenMP
#export OMP_NUM_THREADS=$SLURM_CPUS_PER_TASK

python multithread.py $SLURM_CPUS_PER_TASK
```

```
#!/bin/bash
### Script de soumission SLURM

#SBATCH --nodes 1
#SBATCH --cpus-per-gpu 2
#SBATCH --time 02:00:00
#SBATCH --job-name my_jupyter_notebook
#SBATCH --gres=gpu:1
#SBATCH -p gpu-intel
#SBATCH -t 4:00:00

module load anaconda/2020.11

jupyter notebook --no-browser --port=8081 --ip=$(hostname -s)
```

- Important : si vous avez beaucoup d'écritures privilégiez de travailler sur le disque local des nœuds de calcul

```
#!/bin/bash
### Script de soumission SLURM

#SBATCH -J MON_JOB
#SBATCH -p cpu
#SBATCH --ntasks=64 --nodes=1
#si on veut repartir les process MPI
##SBATCH --nodes=2 --ntasks-per-node=32
#SBATCH -t 0:30:00

SCRATCH=/scratch/$USER/$SLURM_JOB_ID
echo "creation du repertoire $SCRATCH"
srun -m arbitrary -w $SLURM_NODELIST mkdir -p $SCRATCH
echo "copie du home vers local"
srun -m arbitrary -w $SLURM_NODELIST cp -r $SLURM_SUBMIT_DIR/*  $SCRATCH
cd $SCRATCH

module load gcc/10.2 openmpi/4.1
#compilation
make

#pas besoin de fichier machinefile ou hostfile
#pas besoin d indiquer le nombre de process mpi (option np) cf. $SLURM_NTASKS
mpirun mpi_write

echo "copie du local vers home"
srun -m arbitrary -w $SLURM_NODELIST cp -r $SCRATCH/out_* $SLURM_SUBMIT_DIR
srun -m arbitrary -w $SLURM_NODELIST rm -rf $SCRATCH
```

- N'hésitez pas à nous contacter pour vous faire accompagner

Géosciences pour une Terre durable
brgm

MONITORER VOS JOBS

- squeue --start -u $USER
  - Vos jobs, avec une date prévisionnelle pour vos jobs en attente

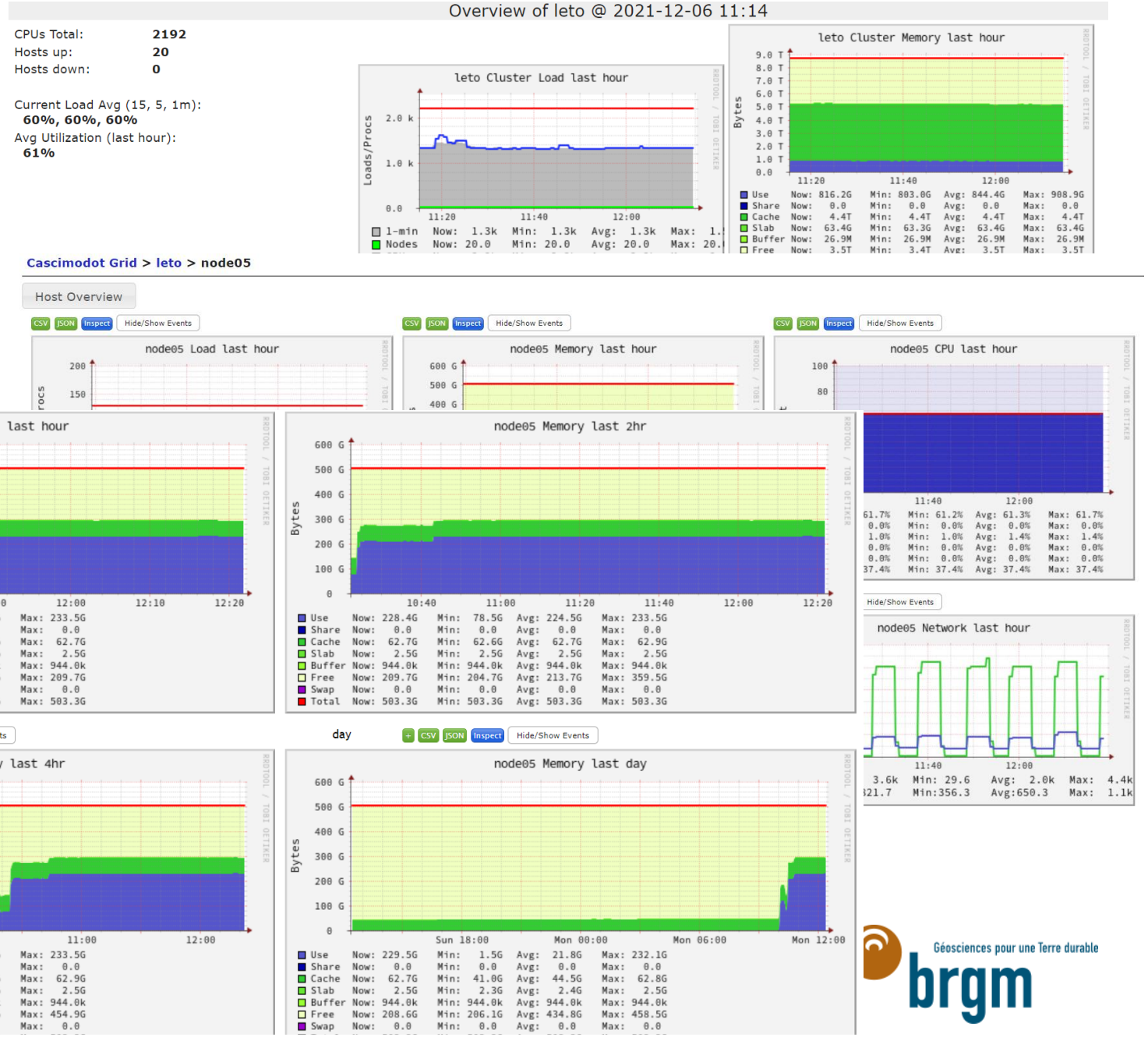
```
fboulahya@leto:~/JourneeCascimodot> scontrol show jobid -dd 23927
JobId=23927 JobName=JOB_MULTITHREAD
   UserId=fboulahya(1003) GroupId=fboulahya(1003) MCS_label=N/A
   Priority=8813 Nice=0 Account=brgm QOS=normal
   JobState=COMPLETED Reason=None Dependency=(null)
   Requeue=1 Restarts=0 BatchFlag=1 Reboot=0 ExitCode=0:0
   DerivedExitCode=0:0
   RunTime=00:05:01 TimeLimit=00:30:00 TimeMin=N/A
   SubmitTime=2021-12-07T12:23:09 EligibleTime=2021-12-07T12:23:09
   AccrueTime=2021-12-07T12:23:09
   StartTime=2021-12-07T12:23:33 EndTime=2021-12-07T12:28:34 Deadline=N/A
   SuspendTime=None SecsPreSuspend=0 LastSchedEval=2021-12-07T12:23:33
   Partition=cpu AllocNode:Sid=leto:10285
   ReqNodeList=(null) ExcNodeList=(null)
   NodeList=node03
   BatchHost=node03
   NumNodes=1 NumCPUs=16 NumTasks=1 CPUs/Task=16 ReqB:S:C:T=0:0:*:*
   TRES=cpu=16,mem=32G,node=1,billing=16
   Socks/Node=* NtasksPerN:B:S:C=0:0:*:* CoreSpec=*
   JOB_GRES=(null)
     Nodes=node03 CPU_IDs=56-63,120-127 Mem=32768 GRES=
   MinCPUsNode=16 MinMemoryCPU=2G MinTmpDiskNode=0
   Features=(null) DelayBoot=00:00:00
   OverSubscribe=OK Contiguous=0 Licenses=(null) Network=(null)
   Command=/home/fboulahya/JourneeCascimodot/multithread.job
   WorkDir=/home/fboulahya/JourneeCascimodot
   StdErr=/home/fboulahya/JourneeCascimodot/slurm-23927.out
   StdIn=/dev/null
   StdOut=/home/fboulahya/JourneeCascimodot/slurm-23927.out
   Power=
   NtasksPerTRES:0
```

```
fboulahya@leto:~> sacct -j 23927  --format=User,JobID,Jobname,partition,state,time,start,end,elapsed,MaxRss,nnodes,ncpus,nodelist
     User        JobID    JobName  Partition      State  Timelimit               Start                 End    Elapsed     MaxRSS   NNodes      NCPUS        NodeList
--------- ------------ ---------- ---------- ---------- ---------- ------------------- ------------------- ---------- ---------- -------- ---------- ---------------
fboulahya 23927        JOB_MULTI+        cpu  COMPLETED   00:30:00 2021-12-07T12:23:33 2021-12-07T12:28:34   00:05:01                     1         16          node03
          23927.batch       batch             COMPLETED            2021-12-07T12:23:33 2021-12-07T12:28:34   00:05:01     16144K        1         16          node03
          23927.extern     extern             COMPLETED            2021-12-07T12:23:33 2021-12-07T12:28:34   00:05:01      1504K        1         16          node03
```

MONITORER VOS JOBS

Etat du cluster

- Ganglia
  - Monitoring des nœuds pendant les calculs pour voir consommation CPU,RAM,réseau,…



Géosciences pour une Terre durable
brgm

SUPPRIMER DES JOBS

- scancel jobid
- scancel -u $USER

A retenir

- Décrire au mieux son job avec a minima une durée réaliste
- Dès que vous avez beaucoup d'écritures privilégiez de travailler sur le disque local des nœuds de calcul
- N'hésitez pas à nous contacter pour qu'on vous accompagne